Ball and Beam Control Project
ES245: Control Systems

Aditya Prasad 22110018, Akshat Pratap Singh 22110023
Nikhil Kumar 22110166, Tapananshu Gandhi 22110270
Mechanical Engineering Department
IIT Gandhinagar
Palaj, India

***Abstract*—The ball and beam system is an example of an unstable dynamic system commonly used in control theory applications. This report aims to design a suitable controller to stabilize the ball's position, ensuring precise control in various operational scenarios.**

***Index Terms*—ball and beam, control systems, PID controller, system dynamics**

I. Introduction

The ball and beam system is an example of an unstable dynamic system commonly used in control theory applications. This system involves a ball that must be positioned on a beam, which a motor can tilt. The objective is to automatically regulate the ball's position by adjusting the angle of the beam in response to its movement. Given the system's inherent instability, effective control techniques are essential to maintain the desired position of the ball, particularly in applications such as aerospace stabilization and process control in industrial settings. This report aims to design a suitable controller that accurately calculates and adjusts the beam's angle to stabilize the ball's position, ensuring precise control in various operational scenarios.

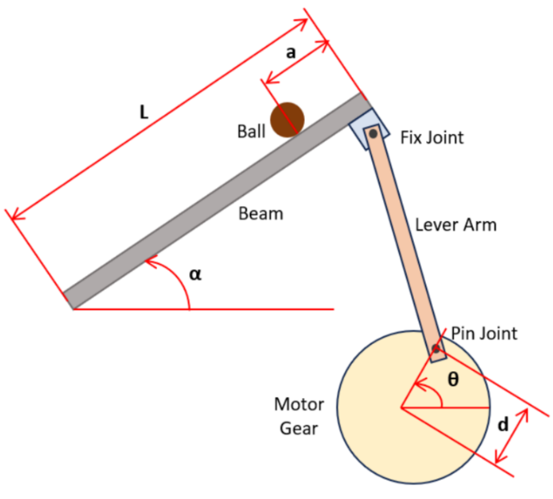

Fig. 1. Physical System

II. Tasks

Task 1: Define the System

1.1 System Dynamics Equations: The following Lagrangian equation can describe the motion of the ball:

$$0 = \left(\frac{J}{R^2} + m\right)\ddot{r} + mg\sin\alpha - mr\dot{\alpha}^2 \tag{1}$$

For the sake of simplification, we will disregard the influence of the second derivative of the input angle α on the second derivative of r.

1.2 Transfer Function: Taking the Laplace transform of the equation yields:

$$\left(\frac{J}{R^2} + m\right)R(s)s^2 = -mg\frac{d}{L}\Theta(s) \tag{2}$$

By rearranging this equation, we can derive the transfer function from the gear angle $\Theta(s)$ to the ball position $R(s)$:

$$P(s) = \frac{R(s)}{\Theta(s)} = -\frac{mgd}{L\left(\frac{J}{R^2} + m\right)}\frac{1}{s^2} \qquad [\frac{m}{rad}] \tag{3}$$

It is important to note that this transfer function represents a double integrator, indicating marginal stability and presenting a challenging control problem.

1.3 Linearization: The linearized system equations can be expressed in state-space form. By selecting the ball's position (r) and velocity ($\dot{r}$) as state variables, with the gear angle (θ) as the input, the state-space representation is given by:

$$\begin{bmatrix} \dot{r} \\ \ddot{r} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} r \\ \dot{r} \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{mgd}{L\left(\frac{J}{R^2}+m\right)} \end{bmatrix}\theta \tag{4}$$

Task 2: Analysis of the System

2.1 MATLAB Analysis: We assume suitable values for the parameters in the transfer function, including mass of the ball (m), radius of the ball (R), lever arm offset (d), length of the beam (L), ball's moment of inertia (J), ball's position on beam (a), beam angle, and servo gear angle.

- Plot poles/zeros.

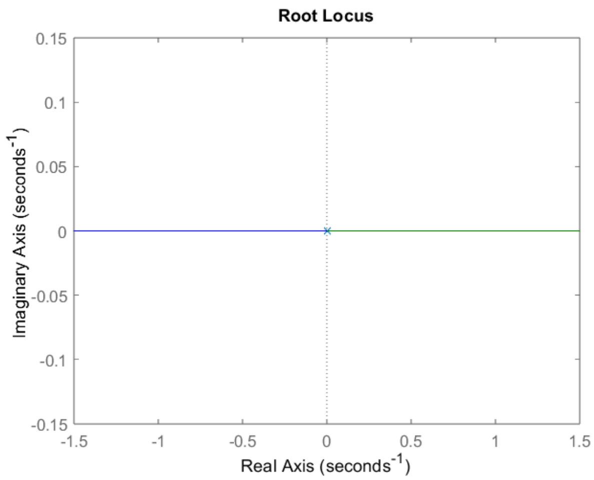

Fig. 2. Root Locus

We can observe double poles at the origin. There are two root locus branches originating at the origin, each of which goes towards zeros located at positive and negative real infinity, respectively.

- Plot open-loop step response (servo gear angle in 1-radian step).

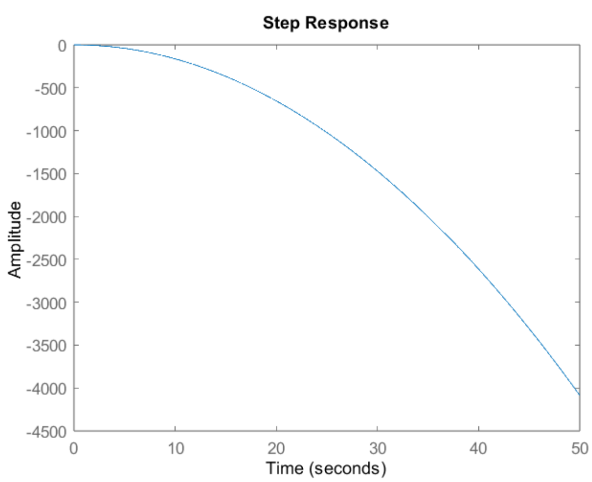

Fig. 3. Unit Step response plot

It can be clearly seen from the above open-loop step response that the step response is monotonically diverging away from its intended value. This necessitates a control system to control this monotonically increasing deviation.

2.2 System Type Identification: Since the Open-loop Transfer function has an s-squared term in the denominator, hence this is a second-order and Type-II system.

Task 3: PID Control

3.1 Controller Design & 3.2 Observations: Design a controller with unity feedback and plot the performance of the following controllers for varying gains and mention your observations:

- **Proportional controller.**

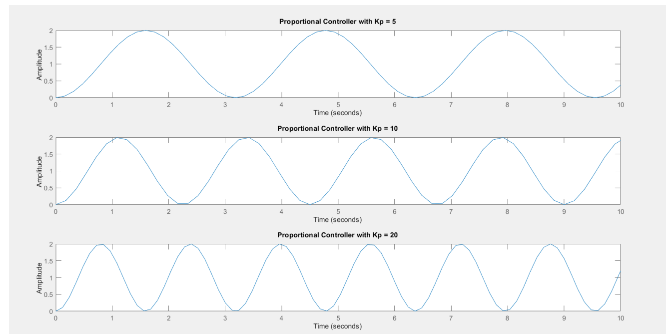

Fig. 4. Unit Step response for various proportional controllers

The step response using proportional controllers is within a bounded range. This is a positive development as compared to the step response without using any controller.

From the above figure, it can be observed that as we increase the proportional gain value of the controller, the step response becomes more oscillatory in nature.

- **Proportional-derivative controller.**

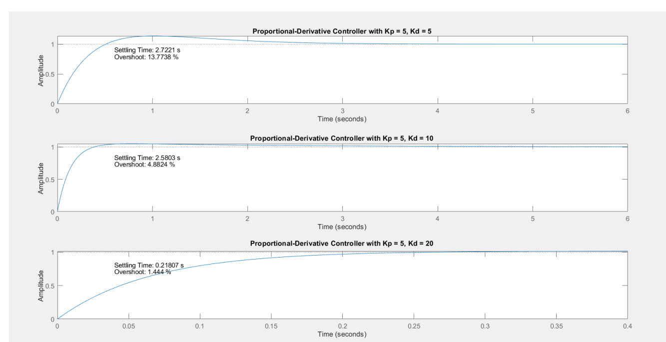

Fig. 5. Unit Step response for various PD controllers

The step response using proportional-derivative controllers is within a bounded range and non-oscillatory damped in nature. This is better than the response using a proportional controller.

From the above figure, it can be observed that as we increase the gain of the derivative controller, the system becomes quicker, i.e., the settling time decreases. It can also be observed that as we increase the derivative gain value, the maximum overshoot percentage decreases.

- **Proportional-integral-derivative controller.**

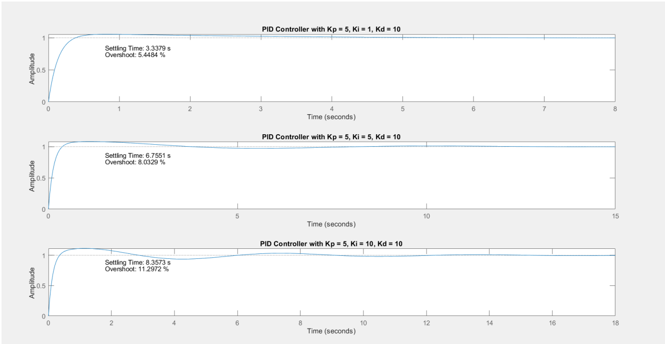

Fig. 6. Unit Step response for various PID controllers

From the above figure, it can be observed that as the value of the integral controller's gain is decreased, the system's settling time and maximum overshoot value decreases.

3.3 Settling Time Criteria: Design a controller to meet the following criteria: settling time less than 3 seconds and overshoot less than 5%.

We observed that low Kp values rendered a non-oscillatory system response. It was also observed that a large Kd value gave us a response with a low overshoot percentage and low settling time. It was observed that a low Ki value gave us a system response with a low overshoot percentage.

We took a hit-and-trial approach to get a suitable combination of controller parameters(Kp, Kd, Ki) to get the controller with the following metrics:

```
Final Performance Metrics:
Overshoot (%): 3.8765
Settling Time (s): 2.4566
Final Tuned PID Gains:
Kp: 4.3422
Ki: 0
Kd: 10.6968
```

Fig. 7. System parameters for the designed controller

The step response for a system with the above-designed controller is:

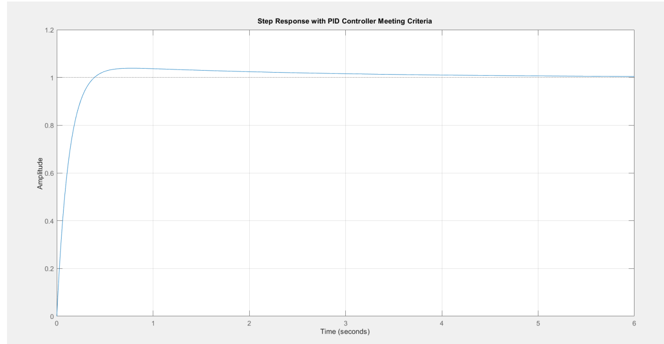

Fig. 8. Step response of the designed PID controller

Task 4: Simulation of Ball and Beam in MATLAB

4.1 Simulink Model: Build the ball beam model in Simulink and generate the system's open-loop response.

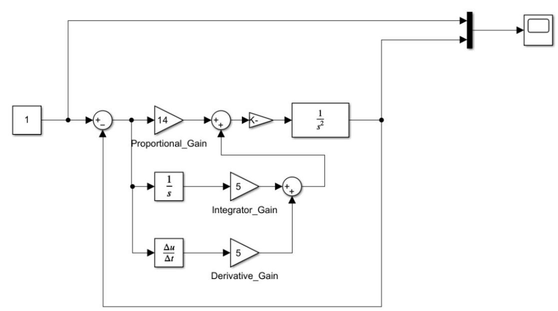

Fig. 9. Simulink model

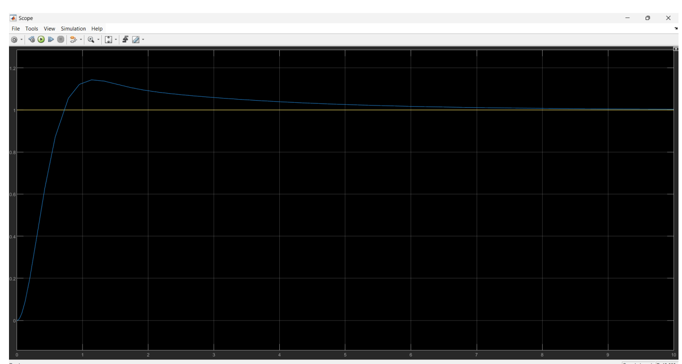

Fig. 10. Open loop Step response of the Simulink model

4.2 Linearization and Compensator Design: Linearize the model and design a compensator to meet the following design criteria: overshoot of less than 5% and settling time of less than 5 seconds. Generate the system's closed-loop response.

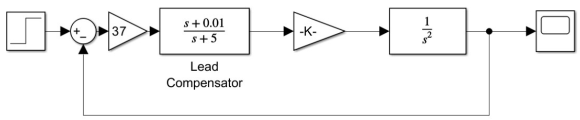

Fig. 11. Simulink model

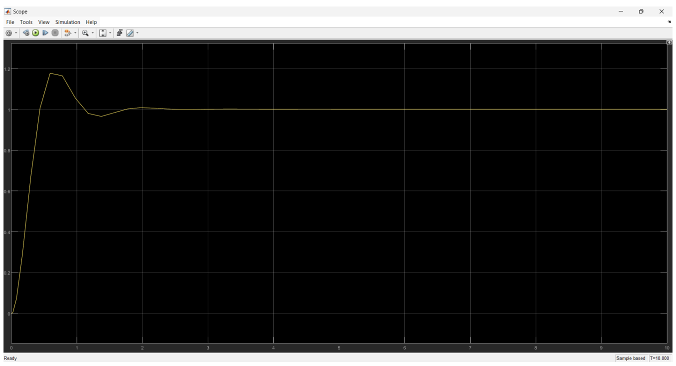

Fig. 12. Closed loop Step response of the Simulink model

4.3 Simscape Model: Develop a Simscape model for the ball beam system.

We choose suitable values of system parameters:

- Mass of ball(m): 2.5 grams
- Radius of ball(R): 20 mm
- Lever arm offset(d): 40mm
- Length of Beam(L): 28 cm
- Ball's Moment of Interia(J): 5.735e-6 kg-m^2

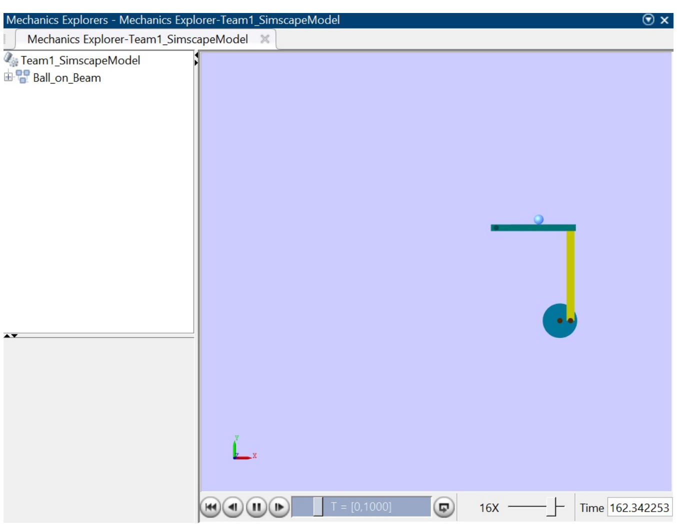

Fig. 13. Simscape model

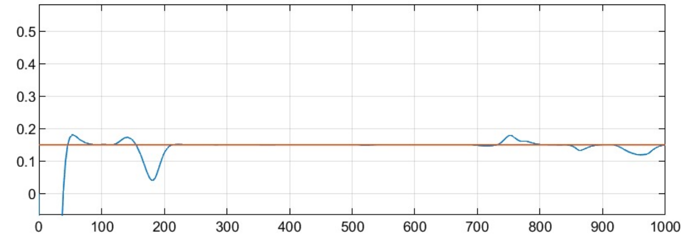

Fig. 14. Step response of the Simscape model

Task 5: Physical System

5.1 CAD Model: Develop a CAD model of the system and build the physical system.

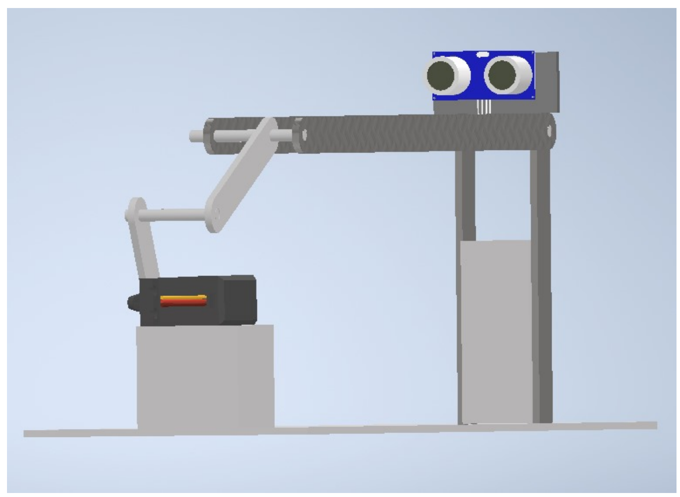

Fig. 15. CAD model

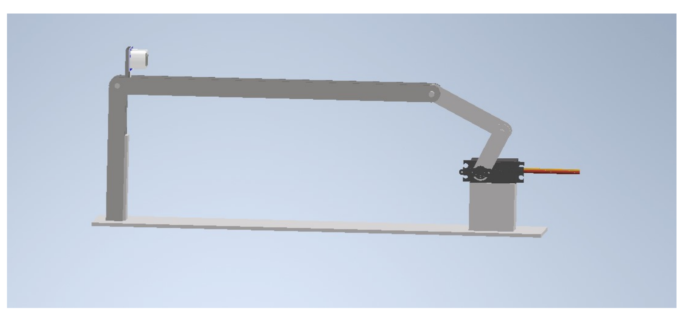

Fig. 16. CAD model

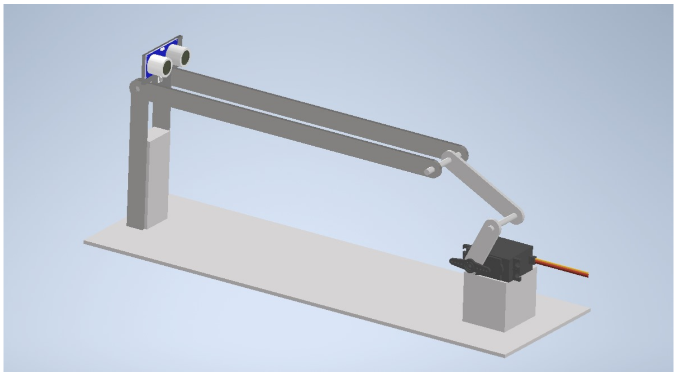

Fig. 17. CAD model

5.2 PID Controller Design: Design a PID controller to stabilize the ball on the beam at a given distance from the end. Comment on your observations and challenges in implementing the PID control.

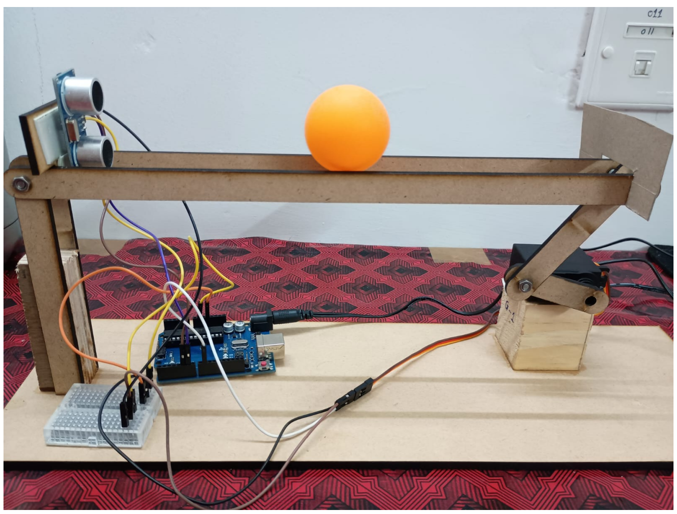

Fig. 18. PID Controller

While implementing the above PID controller we faced numerous challenges as stated below:

Non-Linear System Dynamics: The non-linear system dynamics become harder to model accurately in the real world. Approximating the nonlinear behavior in a linear framework may not yield an accurate model for all operating conditions.

Parameter Tuning: The process of tuning the proportional (Kp), integral (Ki), and derivative (Kd) gains to achieve the desired performance is challenging. Improper tuning can result in issues like oscillations, long settling time, or even system instability

Proportional Gain (Kp): A high proportional gain improves response time but can lead to overshoot and oscillations.

Integral Gain (Ki): The integral term is necessary to eliminate steady-state error, but high integral gain can lead to instability and slower response time due to integral windup.

Derivative Gain (Kd): The derivative term helps reduce overshoot and improves damping, but it is sensitive to noise and may cause erratic behavior if not tuned properly.

Acknowledgment

The authors would like to thank IIT Gandhinagar Mechanical Lab for the resources and guidance provided to complete this project. We want to extend our heartfelt gratitude towards Professor Vineet and PHD Rajdeep Singh for their invaluable help and guidance.

References

[1] ”Ball and Beam System,” Control Tutorials for MATLAB and Simulink, University of Michigan, accessed October 20, 2024. [Online]. Available: https://ctms.engin.umich.edu/CTMS/index.php?example=BallBeam&section=SimulinkSimscape